Towards a maintenance semantic architecture.

Mohamed Hedi Karray, Brigitte Chebel-Morello, Noureddine Zerhouni

TOWARDS A MAINTENANCE SEMANTIC ARCHITECTURE

Mohamed Hedi Karray[a], Brigitte Chebel Morello[a] and Noureddine Zerhouni[a]

[a] *Automatic Control and Micro-Mechatronic Systems Department, 24, Rue Alain Savary, 25000 Besançon, France*

Technological and software progress with the evolution of processes within company have highlighted the need to evolve systems of maintenance process from autonomous systems to cooperative and sharing information' system based on software platform. However, this need gives rise to various maintenance platforms. The first part of this study investigates the different types of existing industrial platforms and characterizes them compared to two criteria namely: information exchange and relationship intensity. This allowed identifying the e-maintenance architecture as the current most efficient architecture. Despite its effectiveness, this latter can only guarantee technical interoperability between various components. Therefore, the second part of this study proposes a semantic-knowledge based architecture, thereby ensuring a higher level of semantic interoperability. To this end, specific maintenance ontology has been developed.

Key words: maintenance systems, e-maintenance, interoperability, semantic maintenance, ontology.

1 INTRODUCTION

Today's enterprise must respond to increasingly demands in terms of quality and quantity of products and services, responsiveness and costs reducing. To deal with these demands, company must have a reliable production system, well maintained by an efficient and inexpensive maintenance system. A performance and well-organized maintenance service contributes to the production system consistency, it will extend the life of industrial equipment and thus the best overall performance throughout the company. This need for maintenance concerns any type of enterprise, industry or service provider.

Since the 80'Th, a phase of maintenance services structuring and standardization is being established. Markets' evolution, globalization and their emphasis on profit and competitiveness of the firm cause the development of new concepts of production organization as well as the maintenance organization. At the same time, the quality aspect is beginning to play an important role as well as the dependability and, specifically, the maintenance function in companies.

New technologies of information and communication technologies (ICT) have helped to establish and evolve these roles. Thanks to ICT, Web emergency and Internet, the achievement of maintenance services and monitoring can be performed automatically, remotely and through various distributed information systems. Hence the emergence of the concept of services offered through maintenance architectures, ranging from autonomic systems to integrated systems where cooperation and collaboration are vital to any operation.

On the other hand, the set up of these fundamental aspects is a complex task. Thus we are particularly interested in the type of exchanged information, and the complex relationships between different systems and applications in these architectures. At this level, we are confronted to a classical problem in information systems, which is interoperability. This latter means the ability of two or more systems or components to exchange information and to use the information that has been exchanged [1].

In this paper we focus on the semantic interoperability of this exchanged information and how it can grantee an understandable exchange and to evolve existing architecture form static architecture to an intelligent one based on knowledge. To make that, we build a maintenance ontology, which will be shared between different systems of the architecture.

The objective of this paper is twofold: (i) recense existing maintenance architectures and (ii) propose a new generation of maintenance architecture semantically interoperable.

The rest of the paper is organised as follow. Section 2 is devoted to present the complex characteristics of systems and its relations. Section 3 and 4 are devoted to tale maintenance system historic and various existed maintenance architectures. We present in section 5 the semantic interoperability problematic and its importance to set up the s-maintenance architecture. In the next section 6 we build a domain ontology of maintenance based on the analysis of the maintenance processes. Future work about our ontology use and evaluation and the conclusion are developed consequently in sections 7 and 8.

2 COMPLEX SYSTEMS CHARACTERISTICS

We develop in this section two classification criteria for characterizing software architectures from a macroscopic view freeing details to be studied (the protocols ...) when we want to improve these architectures, including the e-maintenance.

2.1 Information evolution

The information used in different applications in the field of maintenance has changed in the light of information technology developments and depending on the complexity of the industrial environment.

In the past, this information has been manually entered on paper (drawings, diagrams, manuals) and was verbally exchanged between operators in an informal way. Unlike today, the information is different. It has become formalized and structured to be manipulated by information systems. At the same time, enterprise environment becomes increasingly complex and production systems are becoming more dynamic, which makes the context of the information's use more variable and unstable. Information is uncertain; it evolves with the changing context. The way to reduce this uncertainty is through the implementation of this information in a context with meaning and direction, by turning it into knowledge in a given objective. This knowledge then becomes, along with other information and knowledge a source to acquire skills. Today's information systems handle this knowledge to provide a decision support for its users on problem solving and to improve their skills in this field.

2.2 Relations between systems

Thanks to technological and informatics evolution, information systems which were independent and autonomous begin to cooperate by exchanging and sharing information. More recently, new information technologies and communication (ICT) have enabled the migration of these different systems into an integrated system where cooperation and collaboration are essential to any operation. There are different types of relationships between systems under review and will be the basis for the classification of different architectures in maintenance (see Figure 1).

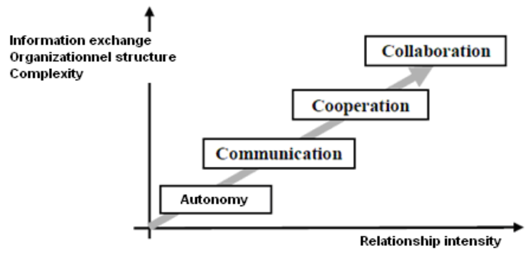

Figure 1. Relationship intensity between systems.

- Autonomy relationship is a regime under which a system has the maximum power of management and is independent of all other systems and components. There is communication between systems and it must be self-sufficient in terms of necessary information.

- Communication relationship is a link between two or more systems that allows transfers or exchanges. The information transmitted in the communication is no longer limited to alphanumeric characters and also include images, sound and video clips. In this context, the term communication is often used as synonymous with telecommunications.

- The cooperative relationship presents a cooperative work being done by a work division in which each actor is responsible for part of the resolution of the problem. In our context, it is mainly technological and industrial cooperation, therefore, a cooperative agreement between independent systems that are committed to carry out joint production of maintenance services.

- The collaborative relationship is a strategic partnership to achieve excellence through a combination of skills, suppliers or products. Collaboration involves a mutual commitment of stakeholders in a coordinated effort to resolve the problem by pooling resources, information and skills to better adapt to their environment organizations.

3 HISTORY OF MAINTENANCE'S COMPUTING SYSTEM

The development of computer systems in the field of industrial maintenance began when the maintenance has been recognized as a fundamental function in the company and a particular stress was laid on the study and development of procedures of this function.

Information used in maintenance has changed in according to the evolution of information technologies and in according to the growth of enterprise environment complexity. The information structure has changed in order to be handled by information systems.

We can identify various aspects in the evolution of computer system maintenance [2, 3]:

- Computerization of the procedures of maintenance: The automation of the business management allowed computerizing several maintenance procedures. Computer files of equipment, interventions, stocks, plans and diagrams etc were thus created. The integration of these files and the automation of the maintenance activities were possible thanks to CMMS packages (*Computerized Maintenance Management System*). The daily events of maintenance were treated: the blackout, preventive execution, stocks management.

- Interfacing with software packages: Thereafter, these software packages had to interface with the other enterprise software such as purchasing and accounting, already computerized. Large ERP systems (Enterprise Resource Planning) are a next step in streamlining the business processes and integration of maintenance with other corporate functions.

- Evolution of the technical field: Informatics has also made progress in the technical field of maintenance. Modern techniques of analysis of maintenance and control have emerged in parallel computing: vibratory analysis, oil analysis, IR thermography, hot ultrasounds etc. We can distinguish among these systems two main groups: analysis systems and acquisition and control systems.

* Analysis systems, sometimes coupled with expert systems have been developed. The analysis systems are also intended to provide decision support in diagnosis, prognosis and repair of equipment operators, etc…

* Among the acquisition and control systems, we can quote SCADA (supervisory control and data acquisition), command-control of the equipment, technical data and documentation management systems, etc…

- Integration of intelligent modules in maintenance architecture: The presence of these various intelligent modules of maintenance leads us to make them communicate and collaborate. The construction of intelligent modules or bricks must contribute to provide indicators to make the right strategic decision and maintenance policy.

- Development of ICT: The development of new information and communication technologies, the extension of the Internet in the enterprise, application integration, and the emergence of new policies for maintenance indicate a new stage for the computerization of maintenance, that which some call "maintenance Intelligent”.

This leads to cooperative and distributed architectures of maintenance systems communicating between them or on a basis of networks. Implementation of these maintenance architectures can be done using maintenance platforms whose main idea is to offer a maintenance service via internet. Maintenance platforms proposed in Proteus or OSA / CBM projects can serve as examples.

4 DEFINITIONS OF VARIOUS ARCHITECTURES

We propose a terminology characterizing the various computer systems in maintenance and we classify those under two areas: the type of information used in the system and the intensity of a possible relationship with other systems (see Figure 2). More the relation is intense more the systems are connected and integrated and we speak about common architectures to be implemented across platforms. The volume of automatically managed information is concretized by the surface of the square of each system and increases with the intensity of collaboration and also with the complexity of shared information. We note that there is a parallel between our classification of systems and enterprises classification as presented in several work [4].

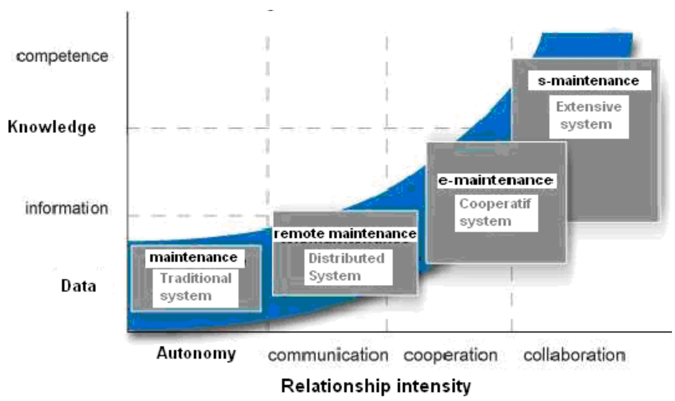

Figure 2. Maintenance architectures classification

- The maintenance system includes a single computer system on this site and used on the site of maintenance. This system is autonomous with data exchange with other systems. In parallel with the classification of companies, this corresponds to the traditional company; therefore we are talking about a traditional architecture of an information system.

- The system of remote maintenance consists of at least two computer systems a transmitter and a receiver of data and information distantly exchanged. According to the definition of AFNOR remote maintenance is "the maintenance of a well executed without physical access of the staff to the equipment". We are talking about a distributed architecture, based on the concept of distance that can transfer data by radio, telephone line or through a local network.

- With the extension of Internet, the systems of remote maintenance emergent towards the concept of E-maintenance. The system of e-maintenance will be implemented on a platform integrating various cooperative distributed systems and maintenance applications. This platform must take support on the global Internet (from where the E-maintenance term) and Web technology allows to exchange, to share and distribute data and information and to create common knowledge. Here the concept intelligent maintenance can be exploited and the proactive and cooperative strategies of maintenance are installed.

- Finally, we propose an architecture intended to improve the performance of the architecture of E-maintenance on the level of the communication and exchange of the data between systems and which makes it possible to take account of the semantics of processed data in the applications - S-maintenance (where " S " means semantic) [5]. We will describe with section 5 this concept which takes support on semantic-knowledge via an ontology of maintenance.

4.1 Maintenance

This is the basic notion where the system is completely autonomous. AFNOR defines it as “combination of all technical, administrative and managerial actions during the life cycle of an item intended to retain it in, or restore it to, a state in which it can perform the required function”.

A maintenance system is represented by an application for maintenance or reliability of the various activities of the maintenance function such as logistics, planning interventions, inventory (managed by the CMMS, ERP), diagnosis and repair (expert systems, databases), monitoring equipment (SCADA, digital control equipment). The architecture of these systems can range according to various objectives. Therefore we propose to describe architectures of these systems by a generic scheme valid for any enterprise system. This scheme consists of two main parts, namely the physical system and management system. This latter produces the whole of the results or decisions based on information coming from the physical system [6]. The acquisition of information is manual or rather limited in its automation and the decisions are thus done by the intermediary of an information system.

4.2 Remote maintenance

The architecture of remote maintenance consists of two or several systems or subsystems apart from each other and exchange data between them. One of the systems can function as a data acquisition system, representing the issuer of structured data. The second system is the receiver, functioning as a data processing system. The transmitter can send data automatically or in response to a request from the receiving system data.

The results of data processing (output) are used by human actors or may be referred to the purchasing system to arrange the data acquisition. So that data can be exchanged and acceptable by both systems they must be structured. Always by keeping the aspect of distance, remote maintenance can be installed on only one production site as it may be distributed among different production or maintenance site and/or a maintenance center.

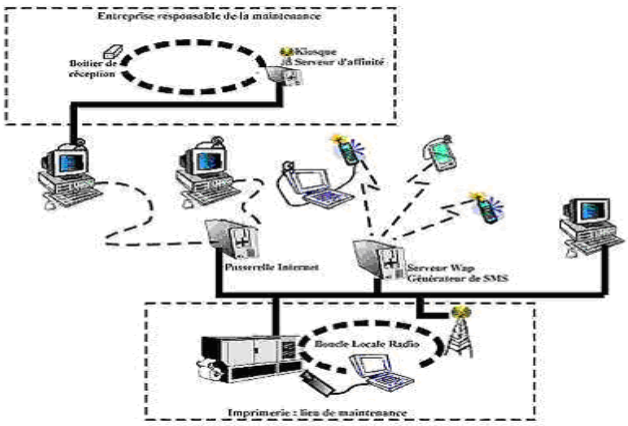

Figure 3. Example of remote maintenance architecture

An example of remote maintenance architecture (cf figure 3) was created in the project TEMIC (Industrial Cooperative Remote Maintenance) which allowed cooperative remote maintenance: not only maintenance staff can perform work at a distance (remote maintenance) but it can do it in collaboration with other experts (cooperative work).

Emphasis was placed on the mobility aspect of the cooperating members on several levels:

- Distant Level: the actors of remote maintenance will be reachable wherever they are via the mobile network (GSM / GPRS).

- Local Level (nomadism): by detection of the presence of remote maintenance actors in a preset perimeter (about 100m) within the company which manages maintenance, to join the most experienced technician on a particular problem.

4.3 E-maintenance

The architecture of e-maintenance is done via Internet that allows to cooperate, exchange, share and distribute information to various partner systems of the network (see figure 4). The principle consists in integrating the whole of the various systems of maintenance in only one information system [7]. Systems offer different formats of information that are not always compatible for sharing; this requires coordination and cooperation between systems to make them interoperable. According to [8], interoperability is "the ability of two communication systems to communicate in an unambiguous way, such systems are similar or different. One can say that making interoperable is creating compatibility". The architecture of e-maintenance must ensure interoperability between each of these different systems.

The project MIMOSA (Machinery Information Management Open Systems Alliance) was the first in 90th years in the United States to develop a complex information system for maintenance management [9]. The project aimed to develop a collaborative network of maintenance by providing the open standard protocol EAI (Enterprise Application Integration). The organization recommends and develops characteristics of information integration to allow the management and the control of added value by the opened, integrated and industry oriented solutions. This latter developed from information blocks to create e-maintenance platform have been proposed in this project [10].

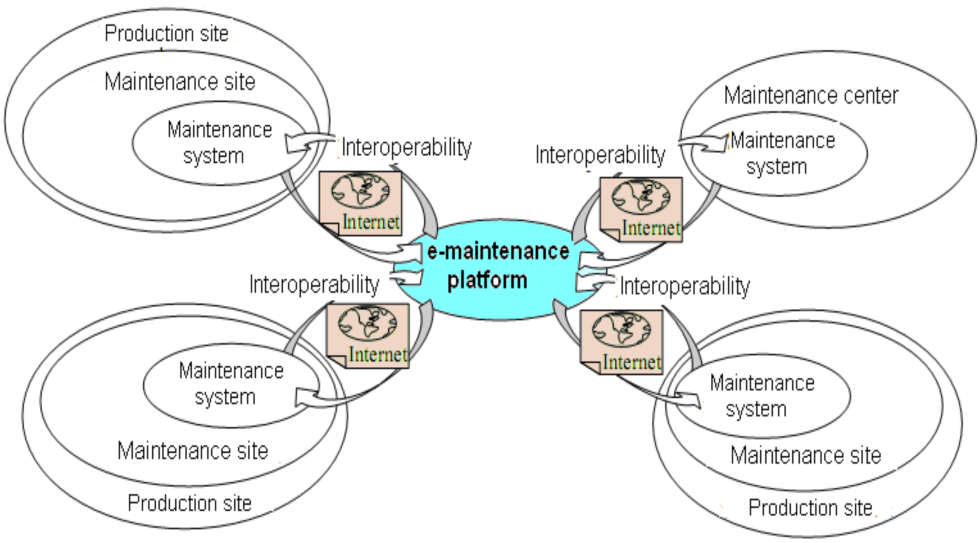

Figure 4. E-maintenance architecture.

A functional architecture OSA / CBM (Open System Architecture for Condition-Based Maintenance) dedicated to the development of strategies for conditional or predictive maintenance [11] was developed from the relational schema MIMOSA CRIS. It contains seven flexible modules whose contents (methodology and algorithms) are configurable by the user (cf figure 5). It can be simplified and adapted to each industrial requirement by reducing modules.

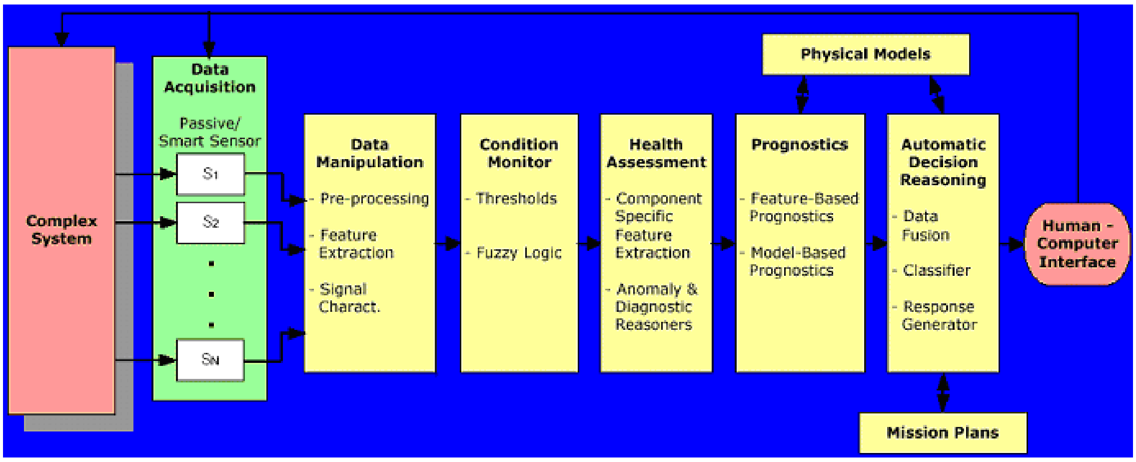

Figure 5. OSA/CBM project

An E-maintenance architecture was presented in the European project Proteus (cf figure 6). The project was designed to provide a cooperative distributed platform of E-maintenance including the existing systems of data acquisition, control, maintenance management, diagnosis assistance, management of documentation, etc. The concept of this platform is defined by a single and coherent description of the installation to maintain, through a generic architecture based on the concepts of Web services and by proposing models and technological solutions of integration. These techniques help to guarantee interoperability of heterogeneous systems to ensure the exchange and sharing of information, data and knowledge. The aim of the platform is not only to integrate existing tools, but also to predict the evolution of these through the introduction of new services.

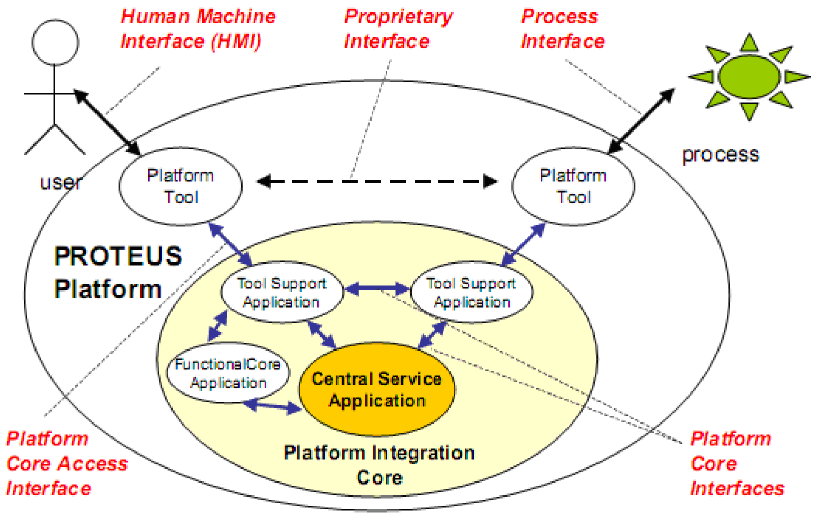

Figure 6. Proteus e-maintenance plateforme[wwww.proteus-iteaproject.com]

Web services were conceived to guarantee interoperability between the various applications of the platform. But they prove that interfaces interconnection protocol does not treat semantics of the output and input data. XML used as bases for data exchange manages structures punts and must be used with the RdF standard to guarantee the bonds between these entities. This architecture guarantees technical interoperability -link between IT systems and services which they provide but does not take account of the semantic interoperability, which consists in giving "Meaning" (semantics) to exchanged information and to make sure that this meaning is distributed in all the interconnected systems. Taking into account this semantics makes possible these systems to combine the information received with other local information and to treat them in a suitable way compared to this semantics [12].

The European project PROMISE (Product Lifecycle Management and Information Tracking Using Smart Embedded Systems) [13] proposes a closed-looped design and lifecycle management system. The objective of PROMISE is to allow information flow management to go beyond the customer, to close the PLC (Product Lifecycle) information loops, and to enable the seamless e-transformation of PLC information to knowledge [14]. This project focused in three working areas related to e-maintenance issues [15]:

- Area 1: E-maintenance and e-service architecture design tools (design of e-maintenance architecture as well as its platform for e-service applications).

- Area 2: Development of watchdog computing for prognostics (development of advanced hashing algorithm for embedded product behaviour assessment and prognostics).

- Area 3: Web-based and tether-free monitoring systems (development of ‘‘interface technologies’’ between the product e-service system platform and Web-enabled e-business software tools).

DYNAMITE (Dynamic Decisions in Maintenance) is an European project which aims to create an infrastructure for mobile monitoring technology and create new devices, which will make major advances in capability for maintenance decision systems incorporating sensors and algorithms [16]. The key features include wireless telemetry, intelligent local history in smart tags, and on-line instrumentation [15].

In [15], Iung et al outline most e-maintenance platforms in order to evaluate their capacity from different points of views as collaboration, process formalization, knowledge management, knowledge capitalization, interoperability, etc… In term of knowledge capitalization the major contribution has especially provided by OSA-CBM and Promise Project platform. This latter has the major contribution in the context of knowledge management too. Regarding interoperability, MIMOSA and OSA-CBM standards have the most relevant contribution in this topic. On the other side, Iung et al didn’t talk about semantic interoperability, and in our knowledge, existing platforms do not focus on this issue. Hence, in this work we stress this problematic and we present s- maintenance (“S” for semantic) architecture which guarantee a high level of semantic interoperability between the various systems of the maintenance platform.

5 SEMANTIC INTEROPERABILITY IN MAINTENANCE ARCHITECTURES

We seek to set up an architecture treating of the semantic interoperability of data.

5.1 Semantic interoperability

The IEEE Standard Computer Dictionary defines interoperability as the “ability of two or more systems or components to exchange information and to use the information that has been exchanged” [1]. From this definition it is possible to decompose interoperability into two distinct components: the ability to exchange information, and the ability to use the information once it has been received. The former process is denoted as ‘syntactic interoperability’ and the latter ‘semantic interoperability’. A small example suffices to demonstrate the importance of solving both problems. Consider two persons who do not share a common language. They can speak to one another and both individuals will recognize that data has been transferred (they can also probably parse out individual words; recognize the beginning and end of message units, etc.). Nevertheless, the meaning of the message will be mostly incomprehensible; they are syntactically but not semantically interoperable. Similarly, consider a person who is blind and one who is deaf, but who both utilize a single language. They can attempt to exchange information, one by speaking and one by writing, but since they are incapable of receiving the messages, they are semantically but not syntactically interoperable [17]. In other words, Semantic interoperability ensures that these exchanges make sense—that the requester and the provider have a common understanding of the “meanings” of the requested services and data [18].

Achieving semantic interoperability among different information systems is very laborious, tedious and error-prone in a distributed and heterogeneous environment [19]. Currently it interests various works which was classified according to Park and RAM [20] in three broad approaches:

1. Cartography interoperability (mapping based approach). It aims to build cartographies between data or elements of models semantically connected [21]. A set of transformation Rules are installed to translate or federate local pattern with a global pattern. One therefore adopts an approach to study semantic interoperability via transformation [22, 23].

2. Interoperability by interoperable languages. These query languages take into account data and metadata to solve semantic conflicts between several data bases interrogation [24].

3. Interoperability through intermediate mechanisms such as mediators or agents. These mechanisms must have a specific knowledge of the area to coordinate different data sources generally via ontologies [25, 26] or via middleware like the Common Object Request Broker Architecture (CORBA) which based on metadata messaging to facilitate interoperability at each level [27].

Other promising ways are presented by Chen et al in [28] who propose two new approaches to resolve semantic interoperability, a model driven interoperability architecture and a Service oriented architecture for interoperability. The model driven interoperability (MDI) architecture based on MDA and enterprise interoperability concepts. The objective of this approach is to allow transforming automatically the models designed at the various abstraction levels of the MDA structure [29]. The service oriented interoperability is based on a Service-Oriented Architectures adopting a federated approach [30], i.e. allowing interoperability of services ‘on the fly’ through dynamic accommodation and adaptation.

As for the above classified approaches, we choose the approach of intermediate mechanisms using ontology engineering.

Indeed, Heiler, Mao et al, Yang et al and others researchers are in agreement that ontology engineering is recognized as the key technology to deal with the semantic interoperability problem [18, 19, 31]. Ontologies specify the semantics of terminology systems in a well defined and unambiguous manner [32], by formally and explicitly representing shared understanding about domain concepts and relationships between concepts. In the ontology based approach, intended meanings of terminologies and logical properties of relations are specified through ontological definitions and axioms in a formal language, such as OWL (Web Ontology Language) [33] or UML (Unified Modelling Language) [34].

This agreement seems promising to us and supports the work of knowledge management which we implemented on a repair and diagnosis module applied to an E-maintenance platform [5]. One of the problems arising from this approach is the definition of common ontology. In our case related to a business approach on maintenance, an ontology relating to the equipments was made during the development of the E-maintenance platform within the framework of European project PROTEUS. The conceived ontology, oriented towards a business approach for the maintenance of industrial equipments, is a common denominator between the various applications implemented in an E-maintenance platform.

However this ontology was not exploited by all assistance modules of the platform, but only by our assistance module of diagnosis and repair. This did not guarantee the semantic interoperability of the platform.

We propose to generalize with the other applications of maintenance assistance, the use of common ontologies in order to guarantee this semantic interoperability. An obstacle to such use is to have a knowledge management approach during the development of the assistance maintenance systems.

5.2 S-maintenance architecture

The architecture of S-maintenance platform takes support on the architecture of E-maintenance where the interoperability of the various integrated systems in the platform is guaranteed by an exchange of knowledge represented by an ontology. So that information sharing in the E-maintenance cooperative network is without difficulty, we are required to formalize this information in a way to be able to exploit it in the various systems belonging to the network. We are extending coordination between network partners and we develop an ontological base of the sharing information. Systems share the semantics created for the common architecture of the E-maintenance platform (cf figure 7). This terminological and ontological base models the whole of knowledge of the domain. It will play the role of a memory to set up a knowledge management and capitalization system and to thus exploit the experience feedback to improve maintenance system functioning. This system will use the knowledge engineering tools as well as knowledge management. The software tool must play the role of service integrator able to be connected to the other systems, specific to companies.

This knowledge system makes possible to identify, capitalize and restore necessary knowledge to control, using a support environment [35]. Semantics has three levels, namely the general concepts of maintenance, application domain concepts, and specific concepts to each company.

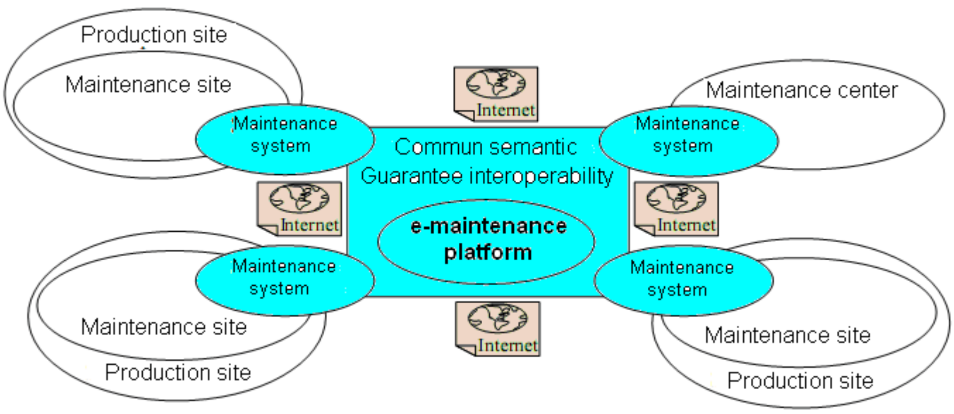

Figure 7. S-maintenance architecture

This system takes support on the concept of E-maintenance with an exchange of information either on the Web services but requiring additional constraints based on standard "OKC" resulting from the semantic Web. The semantics of exchanged information requires the creation of a domain ontology common to the various systems. It allows using and creating knowledge and skills which lead to the use of knowledge management techniques and allows capitalizing acquired knowledge. Systems collaborate, which requires a coordinated effort to solve problems.

6 DOMAIN ONTOLOGY OF MAINTENANCE

Several research works tried to build a maintenance ontology. In the software maintenance area , KITCHENHAM et al in [36] suggest that empirical studies of maintenance are difficult to understand unless the context of the study is fully defined. We developed a preliminary ontology to identify a number of factors that influence maintenance. The purpose of the ontology was to identify factors that would affect the results of empirical studies. We present the ontology in the form of a UML model. Ruiz et al in [37] developed a semi-formal where the main concepts, according to the literature related to software maintenance, have been described. This ontology, besides representing static aspects, also represents dynamic issues related to the management of software maintenance projects. REFSENO (A Representation Formalism for Software Engineering Ontologies) [38] was the methodology used in this work. Matsokis and Kiristis in [39] propose an ontology-based approach for product lifecycle management, as extension of the ontology proposed in Promise project [40]. This latter provide a Semantic Object Model for Product Data and Knowledge Management. The SOM provided a commonly accepted schema to support interoperability when adopted by different industrial partners.

However, these works can be analyzed within two points of view: what does it present? And what does it regard? The two first works try to conceptualize the entire maintenance domain; nevertheless the second two works focus only on the product lifecycle and essentially the middle of life phase [41]. Within the second point of view last works regard to ensure interoperability between industrial partners contrariwise to the first two works which especially aim to ensure the best management of software maintenance and reuse activities.

Thus, we develop a general product maintenance ontology which covers the whole of the maintenance domain having the goal to ensure semantic interoperability among different systems in the maintenance platform.

We take advantages of the classification made by Rasovska et al in the study of maintenance process [42] to set up our ontology. In fact, like shown in figure 8, authors define four fundamental technical and business fields, identified in the general maintenance: (i) equipment analysis which consists of functional analysis and failure analysis; (ii) Fault diagnosis and expertise which aim to help the operator, during his intervention, to diagnose the problem and the prognostic to anticipate the breakdown and to solve it without the recourse to an expert; (iii) Resource management which deals with the resource planning for all maintenance interventions; (iv) Maintenance strategy management which represents decision support concept for maintenance managers.

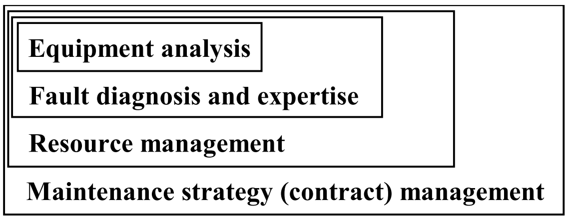

Figure 8. Maintenance process concepts

Based on the study of the maintenance process, dependability concepts and maintenance experts practice, we developed this ontology of maintenance expertise including maintained equipment model associated to maintenance system components as an UML class diagram.

The choice of UML as language of our ontology is based on its graphical expressivity's and the semantics' power recommended in various research works. Cranefield et al in [34] focus on the benefits of using UML as an ontology's language, Bézivin et al in [43] stresses that the meta-models (e.g. UML) in the sense that they are used in the OMG (Object Management Group) address the concept of representation and more specifically to the ontology definition presented in [44].

We have built our own framework, so that from its conception it takes into account the different scope of the maintenance process. This ontology was developed as a tool for sharing semantic between different actors in the e-maintenance platform.

The ontology of the domain, although established independently of the methods of reasoning has a structure which depends on how acquired knowledge will be used for reasoning because experts deliver the knowledge adapted to their reasoning.

The model domain consists of twelve parts (i.e. packages) corresponding to both the structure of the enterprise memory and the maintenance process (see Figure. 9). There are the monitoring management system, site management system, equipment expertise management system, resource management system, intervention management system, the maintenance strategy management model, maintenances management system, equipment states system, historic management system, document management system, functional management system, dysfunctional management system.

The equipment expertise management system is characterized on the one hand, by the equipment components and sub components in a tree form (Component).

Site management system: defined as a unity characterized by emplacement. This site can be a production site which contains operating equipments or a maintenance centre which is the central location to operate and maintain equipments.

Equipment states system: during its operation, equipment may be in one of the following states: Normal state, Degraded state, Failure state, Programmed stop state. We include in programmed stop any stopping of the carried equipment by the authorized personnel. Among scheduled stops, we are interested only to maintenance.

Maintenances management system: this package is related with programmed stop included in the equipment state system. This package manages the different types of maintenance, which are corrective maintenance, conditional maintenance, and preventive maintenance.

The monitoring management system consists of sensors (sensor) installed on the equipment and various measurements (Measure) coming from these sensors. A model of data acquisition (Data acquisition model) manages the acquisition and the exploitation of these measures. This model can trigger the procedure of intervention request according to a threshold measures and is therefore connected with the intervention management model.

The intervention management system focuses on the maintenance intervention. Intervention lets to remedy the equipment failure and is described by an intervention report and characterized by maintenance type.

The maintenance strategies management system is based on technical indicators (Technical indicator) and financial (Financial indicator) for each equipment in a maintenance contract.

The resources management system describes the resources used in the maintenance system, namely human, material, document and their subclasses: operators (Operator), expert (Expert) and manager (Manager) are subclasses of human resource. Tools (Tool), consumables (consumable) and spare parts (Spare part) are subclasses of material resource. The document resource and their subclasses are presented in a separated package.

Document management system: this package presents documentation resources which are indispensable in maintenance as: the equipment plan which contains the design and the model of the equipment and its components, technical documentation where is defined all technical information of an equipment and its use guide, contract which presents maintenance contract, and finally intervention report. This latter is composed by observation, work order, technical comments.

Functional equipment management system functional analysis and associated model (Functional equipment model) characterize the equipment operation by MainFunction and SecondFunction classes. They represent the equipment main and secondary functions to ensure the running smooth of the main function.

Dysfunctional equipment management system each equipment can suffer from breakdowns and failures described in the Failure class and analyzed in the failure Analysis (Failure equipment model). A failure is identified by symptoms (symptoms) caused by origins (Origin) and remedied with a remedial action (Action). It also has characteristics (Characteristics) such as criticality, appearance frequency, non detection and the gravity which are evaluated in the FMECA (*Failure Mode, Effects and Criticality Analysis)*.

Historic management system contains life history which stocks the life historic of an equipment. It is composed by equipments states, interventions and different measurements of the monitoring system.

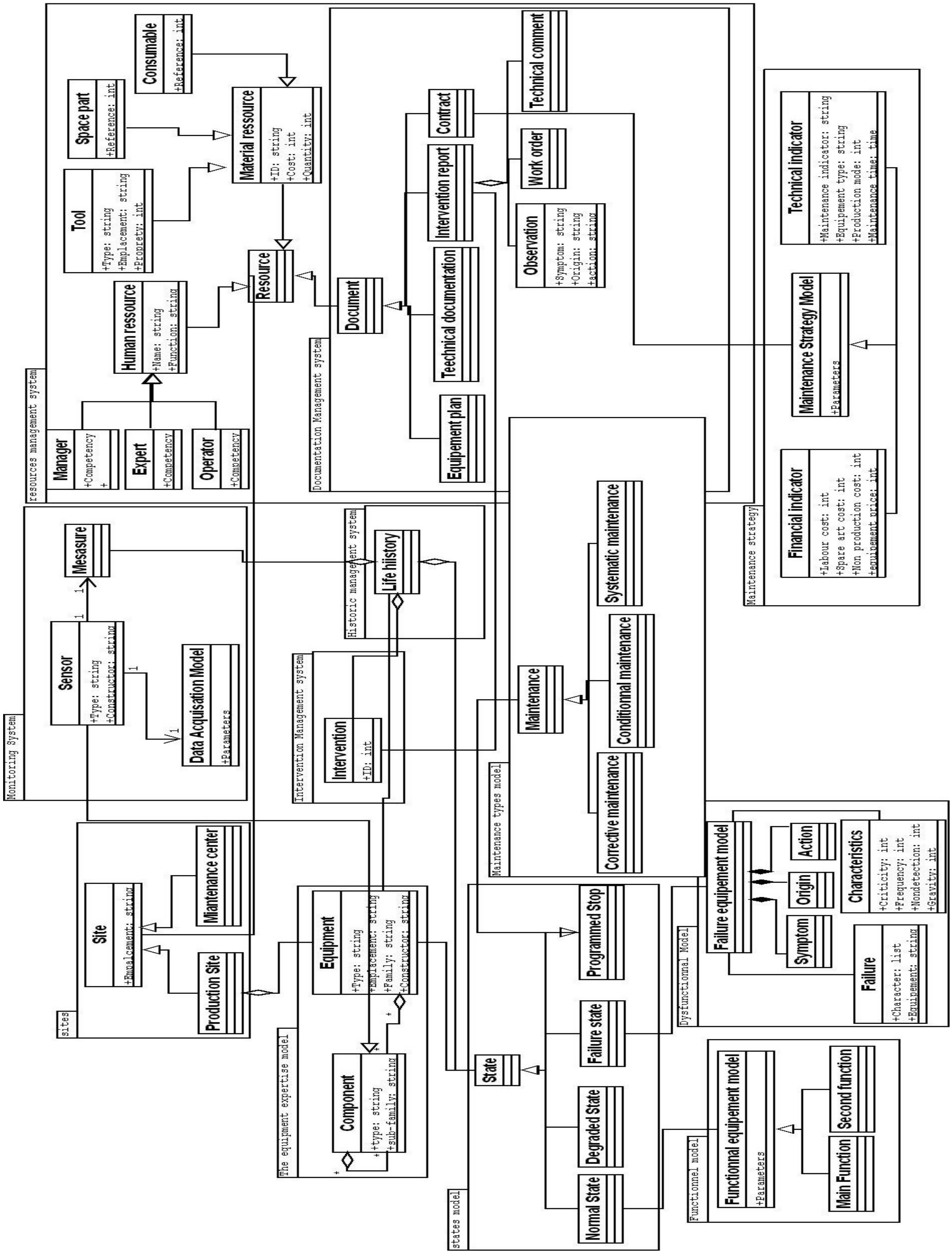

Figure 9. Domain ontology of maintenance

7 FUTURE WORK: ONTOLOGY USE AND EVALUATION

Modelling the domain ontology is very beneficial. But how can we exploit this benefice?; is the question which must be responded. Presenting the ontology via UML class diagram is very beneficial in term of clearness and comprehensibility, but it does not allow the ontology evaluation and the ontology use in the e-maintenance platforms. In other words, we cannot validate the ontology's reasoning, soundness and completeness [45], and we cannot navigate on an UML class diagram. The ontology must be translated to an ontology language allowing reasoning and understandable [46] by the technical components of the platform which use the ontology. Currently we are working to evolve this ontology by the translation on a description logic language to allow the reasoning on the ontology and to implement it throw an interpreted or compiled language. This will permit to us the study of our ontology capacity and quality to guiding us to trails and areas of the ontology development.

In the other hand, we aim to enrich this domain ontology models by adding more concepts and more information to cover all domain areas which can be used to evolve the e-maintenance platform. In the same time we aim to relate this domain ontology with a task ontology providing dynamic activities in the maintenance system as diagnostic, prognostic, detection, acquisition, etc…

8 CONCLUSION

To improve system availability and safety as well as product quality, industries are convinced by the importance role of the maintenance function. Consequently various works are made to evolve this latter. Taking advantages of new information technologies -which allow the integration of various assistance systems via platforms- these works permit to expand and develop maintenance systems.

In this paper we proposed a classification of various existing maintenance architectures to infer a support system architecture for maintenance services. This classification is made according to relations intensity between systems (autonomy, communication, cooperation, collaboration) in a particular architecture. Collaborative or cooperative relation generates a problem on the interoperability level between the architecture systems. The semantic interoperability is considered as one of the complex interoperability's problems that is why we focus on it in this paper. Thus we highlighted the semantic maintenance architecture (S-maintenance) which is based on a common ontology for various systems. Indeed, this developed ontology witnesses a semantic interoperability level. This ontology is related to the maintained equipment and common for the platform to guarantee interoperability between integrated systems and applications. It is based on the maintenance process, dependability concepts and maintenance experts practice, including maintained equipment model associated to maintenance system components as an UML class diagram. The choice of UML as language of our ontology is based on the power graphical expressivity and the semantic of this language. To be operational this class diagram will be translate to an ontology language allowing reasoning like OWL DL language or LOOM.

9 REFERENCES

1. Staff, I.o.E.a.E.E. (1990) IEEE Computer Dictionary: Compilation of IEEE Standard Computer Glossaries 610–1990.
2. Francastel J.C. (2003) Externalisation de la maintenance : Stratégies, méthodes et contrats, Paris : Dunod publication.
3. Boucly F. (1998) Le management de la maintenance : Evolution et mutation, Paris : Afnor Editions.
4. Dedun I & Seville M. (2005) Les systèmes d'information interorganisationnels comme médiateurs de la construction de la collaboration au sein des chaînes logistiques : Du partage d'information aux processus d'apprentissages collectifs. Proceedings of 6th international congress on Industrial engineering. Besançon.
5. Rasovska I., Chebel-Morello B. & Zerhouni N. (2005) Process of s-maintenance: decision support system for maintenance intervention. Proceedings of 10th IEEE International Conference on Emerging Technologies and Factory Automation ETFA'05, Italie.
6. Kaffel H. (2001) La maintenance distribuée: concept, évaluation et mise en oeuvre. Phd thesis, Université Laval, Quebec.
7. Muller A. (2005) Contribution à la maintenance prévisionnelle des systèmes de production par la formalisation d'un processus de pronostic. Phd thesis, Université Henri Poincaré, Nancy.
8. Spadoni M. (2004) Système d'information centré sur le modèle CIMOSA dans un contexte d'entreprise étendue. JESA, Volume 38, n° 5, 497-525.
9. Kahn J. (2003) Overview of MIMOSA and the Open System Architecture for Enterprise Application Integration. Proceeding of COMADEM'03, pp. 661-670. Sweden: Växjö University.
10. Mitchell J, Bond T, Bever K & Manning N. (1998) MIMOSA – Four Years Later. Sound and Vibration, pp. 12-2.

11. Lebold M & Thurston M. (2001) Open standards for Condition-Based Maintenance and Prognostic Systems. Proc. Of 5th Annual Maintenance and Reliability Conference (MARCON 2001), Gatlinburg, USA.
12. Wikipedia 2009. Available on: http://fr.wikipedia.org.
13. Lee J, Ni J. (2004) Infotronics-based intelligent maintenance system and its impacts to closed-loop product life cycle systems. Invited keynote paper for IMS'2004, International conference on intelligent maintenance systems, Arles, France.
14. Kiritsis D. (2004) Ubiquitous product lifecycle management using product embedded information devices. Invited keynote paper of IMS'2004, International conference on intelligent maintenance systems, Arles, France.
15. Muller A, Marquez C & Iung B. (2008) On the concept of e-maintenance: Review and current research. Journal of Reliability Engineering and System Safety 93 1165–1187. Amsterdam: Elsevier Science Publishers
16. Holmberg K, Helle A & Halme J. (2005) Prognostics for industrial machinery availability. POHTO 2005, International seminar on maintenance, condition monitoring and diagnostics. Oulu, Finland.
17. Komatsoulis GA, Warzel DB, Hartel FW, Shanbhag K, Chilukuri R, Fragoso G, de Coronado S, Reeves D M, Hadfield J B, Ludet C & Covitz P A. (2008) caCORE version 3: Implementation of a model driven, service-oriented architecture for semantic interoperability. Journal of biomedical informatics.
18. Heiler S. (1995) Semantic Interoperability. ACM Computing Surveys (CSUR).
19. Mao M. (2008) Ontology mapping: Towards semantic interoperability in distributed and heterogenous environments. PhD thesis University of Pittsburgh.
20. Park J & Ram S. (2004) Information System Interoperability: What Lies Beneath?. ACM Transactions on Information Systems, vol. 22, n°4.
21. Baïna S, Panetto H & Benali K. (2006) Apport de l'approche MDA pour une interopérabilité sémantique. Interopérabilité des systèmes d'information d'entreprise, Processus d'entreprise et SI, RSTI-ISI, pp.11-29.
22. Rahm E & Bernstein P A. (2001) A survey of approaches to automatic schema matching. The International Journal on Very Large Data Bases, vol. 10, n° 4, p.334-350.
23. Halevy A & Madhavan J. (2003) Composing mappings among data sources. Proceedings of the conference on very large databases, p. 572-583, Berlin, Germany.
24. Fauvet MC, Baina S. (2001) Evaluation coopérative de requêtes sur des données semi-structurées distribuées. Proceedings of Information Systems Engineering.
25. Maedche A & Staab S. (2000) Semi-automatic engineering of ontologies from texts. Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering (SEKE 2000), p. 231-239, USA.
26. Halevy G I, Dan Suciu D & Tatarinov I. (2005) Schema mediation for large-scale semantic data sharing. The VLDB Journal, The international Journal on Very Large Data Bases, vol.14, n°1.
27. Tannenbaum, A. (1994) Repositories: potential to reshape development environment. Application Development Trends.
28. Chen D, Doumeingts G, Vernadat F. (2008) Architectures for enterprise integration and interoperability: Past, present and future. Journal of Computers in Industry 59 pp 647–659.
29. Mellor SJ, Scott K, Uhl A & Weise D. (2002) Lecture Notes in Computer Science.
30. ISO 14258. (1999) Concepts and Rules for Enterprise Models. Industrial Automation Systems ISO TC184/SC5/WG1.
31. Yang Q Z & Zhang Y. (2006) Semantic interoperability in building design: Methods and tools. Journal of Computer-Aided Design 38 pp 1099–1112. Amsterdam: Elsevier Science Publishers.
32. Guarino N. (1998) Formal ontology and information systems. Formal ontology and information systems. IOS Press.
33. W3C OWL Web ontology language overview, http://www.w3.org/TR/2003/PR-owl-features-20031215/; 2005 [last accessed 10.05].]
34. Cranefield SJS & Purvis MK. (1999) UML as an ontology modelling language. In Proceedings of the Workshop on Intelligent Information Integration, 16th International Joint Conference on Artificial Intelligence (IJCAI-99).
35. Kramer I. (2003) Proteus: Modélisation terminologique. Technical report INRIA France.
36. Kitchenham B, Travassos G, Von Mayrhauser A, Niessink F, Schneidewind N, Singer J, Takada S, Vehvilainen R & Yang H. (1999) Towards an Ontology of Software Maintenance. Journal of Software Maintenance: Research and Practice 11.

37. Ruiz F, Vizcaino A, Piattini M & García F. (2004) An ontology for the management of software maintenance projects. International Journal of Software Engineering and Knowledge.

38. Tautz C & von Wangenheim C G. (1998) A Representation Formalism for Software Engineering Ontologie. Report of Fraunhofer Institute for Experimental Software Engineering.

39. Matsokis A, Kiritsis D. (2009) An Ontology-based Approach for Product Lifecycle Management. Computers in Industry. Special Issue: Semantic Web Computing in Industry. In press.

40. PROMISE, (2008) FP6 project. www.promise.no.

41. Kiritsis D, Bufardi A & Xirouchakis P. (2003) Research issues on product lifecycle management and information tracking using smart embedded systems. Journal of Advanced Engineering Informatics; 17 (3-4), pp. 189-202 .

42. Rasovska I, Chebel-Morello B & Zerhouni N. (2004) A conceptual model of maintenance process in unified modeling language. Proceedings at 11 th IFAC Symposium on Information Control Problems in Manufacturing 2004 (INCOM)

43. Bézivin J. (2000) De la programmation par objets à la modélisation par ontologie. Journal of Ingénierie de connaissances.

44. Charlet J, Bachimont B, Bouaud J & Zweigenbaum P. (1996). Ontologie et réutilisabilité : expérience et discussion. In AUSSENAC-GILLES N, LAUBLET P & REYNAUD C, Eds., Acquisition et ingénierie des connaissances : tendances actuelles, chapter 4, p. 69–87. Cepaduès-éditions.

45. Uschold M & Gruninger M. (1996) Ontologies: Principles, methods and applications. Knowledge engineering review.

46. Uschold M. (1998) Knowledge Level Modelling: Concepts and Terminology. The Knowledge Engineering Review, vol. 13, N 1.